

Zeitnahmesystem

Nachhaltige Informationstechnologie und Treiberentwicklung

des Master Studiengangs Technische Informatik
an der Hochschule Bochum Campus Velbert Heiligenhaus

von
Jan Küpper

Fachsemester: Sommersemester 2024

Prüfer: Prof. Dr. rer. nat. Peter Gerwinski

Matrikelnummer: 018100841

Ort, Datum: Remscheid, 21.03.2025

Inhaltsverzeichnis

1. Einleitung	3
1.1. Motivation	3
1.2. Aufgabenstellung und Zielsetzung	3
1.3. Projektkontext	3
1.4. Vorgehensweise und Aufbau der Arbeit	3
2. Überblick Zeitnahme bei Radrennen	4
2.1. Arten von Radrennen	4
2.2. Zweck einer Zeitnahme	4
3. Konzeption und Design des Zeitnahmesystem	5
3.1. Anforderungsanalyse und Zielgruppenanalyse	5
3.2. Anwendungsfälle	5
3.3. Systemarchitektur	5
3.4. Wireframes / Mockup	7
3.5. Datenhaltung	7
4. Entwicklung des Zeitnahme-System	10
4.1. Webapplikation	10
4.2. Empfängereinheit	11
5. Nachhaltige Informationstechnologie	13
5.1. Prinzipien der Nachhaltigkeit	13
5.2. Lizenzen und Entscheidung für Open Source	14
5.3. Datenschutz und Sicherheit	14
5.4. Qualitätssicherung	15
6. Fazit und Ausblick	17
7. Anhang	18
A.1 Anwendungsfalldiagramm	18
A.1 Wireframe Seite “Veranstaltungsübersicht”	19
Literatur- und Quellenverzeichnis	20
Abkürzungsverzeichnis	21
Abbildungsverzeichnis	22
Eidesstattliche Erklärung	23

1. Einleitung

Die präzise Zeitmessung ist ein wichtiger Bestandteil von Radrennen. Diese Arbeit beschäftigt sich mit der Entwicklung eines automatisierten Zeitnahmesystems, das nicht nur die Effizienz der Zeitmessung steigert, sondern auch den organisatorischen Aufwand für Veranstalter reduziert und gleichzeitig eine nachhaltige Nutzung von Ressourcen fördert. Ziel ist es, ein System zu schaffen, das den Anforderungen von Radrennen gerecht wird und den Veranstaltern eine zuverlässige, fehlerfreie und zeitgemäße Lösung bietet.

1.1. Motivation

Heutzutage ist die Entwicklung effizienter und nachhaltiger Technologien ein wesentlicher Bestandteil vieler Geschäftsprozesse und Dienstleistungen. Traditionelle manuelle Verfahren zur Zeitnahme bei Radrennen stoßen oft an ihre Grenzen, da sie nicht nur anfällig für Fehler sind, sondern auch einen hohen Zeit- und Ressourcenaufwand erfordern. Die Automatisierung der Zeitnahme ist daher von besonderer Bedeutung, um eine präzise, fehlerfreie und schnelle Zeitmessung zu gewährleisten. Ein automatisiertes Zeitnahmesystem kann nicht nur die Rennzeiten der Teilnehmer effizient und genau erfassen, sondern ermöglicht auch eine Echtzeitanzeige der Ergebnisse für Teilnehmer und Zuschauer.

1.2. Aufgabenstellung und Zielsetzung

Ziel dieses Projekts ist die Entwicklung eines umfassenden und benutzerfreundlichen Zeitnahmesystems für Radrennen, das verschiedene Rennformate unterstützt und präzise und aktuelle Informationen zur Verfügung stellt. Das System soll Zeiten automatisch erfassen, speichern und in Echtzeit anzeigen. Zusätzlich wird angestrebt, den Veranstaltern eine Plattform zu bieten, über die sie Teilnehmer verwalten und Renneinstellungen konfigurieren können.

1.3. Projektkontext

Das Projekt richtet sich an Veranstalter von Radrennen, die auf eine präzise und effiziente Zeitnahme angewiesen sind, sowie an die Teilnehmer, denen die Rennergebnisse zur Verfügung gestellt werden. Ziel ist es, die organisatorische Belastung der Veranstalter zu verringern und gleichzeitig die Teilnehmererfahrung durch eine optimierte Ergebnisdarstellung zu verbessern.

1.4. Vorgehensweise und Aufbau der Arbeit

Die vorliegende Arbeit beginnt mit einer Analyse der bestehenden Zeitnahmesysteme und den spezifischen Anforderungen im Radrennsport. Im Anschluss wird das Konzept und Design des neuen Systems detailliert beschrieben. Darauf folgen die Implementierung und die Integration der verschiedenen Software- und Hardwarekomponenten. Zum Abschluss werden die durchgeführten Tests und das erhaltene Benutzerfeedback diskutiert, bevor ein Fazit gezogen und ein Ausblick auf mögliche Weiterentwicklungen gegeben wird.

2. Überblick Zeitnahme bei Radrennen

Die Zeitnahme spielt eine zentrale Rolle im Radrennsport, da sie nicht nur den Verlauf eines Rennens dokumentiert, sondern auch die Grundlage für die Ergebnisse und die Rangordnung der Teilnehmer bildet. Um den unterschiedlichen Anforderungen der Rennformate gerecht zu werden, ist es notwendig, ein flexibles und präzises Zeitnahmesystem zu entwickeln. In diesem Kapitel werden die verschiedenen Arten von Radrennen vorgestellt, die jeweils spezifische Anforderungen an die Zeitmessung stellen. Zudem wird der Zweck und die Bedeutung einer effizienten Zeitnahme für die Organisation und Durchführung von Radrennen erläutert.

2.1. Arten von Radrennen

Im Radrennsport gibt es verschiedene Rennformate, die jeweils unterschiedliche Anforderungen an die Zeitnahme stellen. Zu den gängigen Formaten zählen Rundenrennen mit Zeitlimit, Rundenrennen mit Rundenziel und Start-Ziel-Rennen. Bei einem Rundenrennen mit Zeitlimit versuchen die Fahrer, innerhalb eines festgelegten Zeitraums möglichst viele Runden zu absolvieren. In einem Rundenrennen mit Rundenziel gilt es, eine vorgegebene Anzahl von Runden zu bewältigen. Beim Start-Ziel-Rennen wird die Zeit einmalig beim Überqueren der Ziellinie erfasst. Ein modernes Zeitnahmesystem muss flexibel genug sein, um diese unterschiedlichen Anforderungen präzise und effizient zu erfüllen.

Radrennen lassen sich außerdem in verschiedene Kategorien unterteilen, Straßenrennen, die auf öffentlichen Straßen ausgetragen werden, Bahnrennen, die auf ovalen Bahnen stattfinden, Mountainbike-Rennen, die auf unbefestigten Wegen oder im Gelände gefahren werden und Zeitfahren, bei denen Einzelstarter gegen die Uhr antreten. Jede Rennart stellt spezifische Anforderungen an die Zeitnahme. Bei Rennen mit Start- und Zielpunkten an verschiedenen Orten ist eine besondere Logistik gefragt, um die Zeit an verschiedenen Messpunkten korrekt zu erfassen. Rundstreckenrennen, bei denen eine festgelegte Rundenzahl oder möglichst viele Runden in einer bestimmten Zeit gefahren werden sollen, stellen besondere Herausforderungen an das Zeitnahmesystem. Es muss die Rundenzeiten präzise erfassen und die gefahrenen Runden exakt zählen.

2.2. Zweck einer Zeitnahme

Ein effektives Zeitnahmesystem ist entscheidend für die Optimierung der Kommunikation und Organisation bei Radrennen. Es ermöglicht Veranstaltern, Rennzeiten präzise zu erfassen und unmittelbar auszuwerten, was die Koordination des Rennverlaufs erleichtert und zur Zufriedenheit der Teilnehmenden beiträgt. Darüber hinaus reduziert ein solches System den administrativen Aufwand erheblich und steigert die Transparenz der Rennergebnisse, was sowohl für die Veranstalter als auch für das Publikum von großem Vorteil ist.

3. Konzeption und Design des Zeitnahmesystem

Das Kapitel beschreibt das Design und die konzeptionellen Überlegungen hinter dem Zeitnahmesystem für Radrennen. Ziel ist es, ein System zu entwickeln, das sowohl den Bedürfnissen der Veranstalter als auch der teilnehmenden Fahrer gerecht wird. Die Grundlage dieses Systems bildet eine präzise und automatisierte Erfassung der Rennzeiten, eine transparente Anzeige der Ergebnisse in Echtzeit sowie eine benutzerfreundliche Verwaltung der Teilnehmer. Im weiteren Verlauf des Kapitels werden die Anforderungen und Zielgruppen des Systems analysiert, die Systemarchitektur mittels UML-Diagrammen dargestellt und die Datenflüsse innerhalb des Systems beschrieben. Ziel ist es, ein robustes, skalierbares und benutzerfreundliches Zeitnahmesystem zu schaffen, das den Anforderungen verschiedener Rennformate gerecht wird.

3.1. Anforderungsanalyse und Zielgruppenanalyse

Das Zeitnahmesystem ist auf zwei Hauptzielgruppen ausgerichtet. Die Veranstalter von Radrennen und die teilnehmenden Fahrer. Für die Veranstalter bestehen die zentralen Anforderungen an das System in der automatischen und präzisen Erfassung der Rennzeiten, der Anzeige der Ergebnisse in Echtzeit, einer benutzerfreundlichen Teilnehmerverwaltung, der Unterstützung verschiedener Rennformate sowie einer hohen Zuverlässigkeit und Bedienfreundlichkeit. Die Teilnehmer wiederum profitieren von der Bereitstellung präziser und aktueller Informationen zu ihren Rennzeiten und Platzierungen, wodurch ihre Rennerfahrung verbessert wird.

3.2. Anwendungsfälle

In der initialen Entwicklungsphase wurden UML-Diagramme erstellt, um die Anforderungen, die Systemarchitektur sowie die Interaktionen der einzelnen Komponenten klar zu definieren. Das Anwendungsfalldiagramm ist im Anhang A.1 dargestellt, es stellt die Anforderungen an das Zeitnahmesystem aus der Perspektive der Veranstalter und Teilnehmer dar.

Aus der Sicht der Veranstalter sollte das Zeitnahmesystem verlässlich, präzise und benutzerfreundlich sein. Es muss die Verwaltung von Veranstaltungen und Teilnehmern unterstützen, inklusive der Erstellung und Bearbeitung von Veranstaltungen, der Anpassung der Teilnehmerliste sowie der Konfiguration und des Starts der Zeitnahme.

Für die Teilnehmer liegt der Schwerpunkt auf Genauigkeit und Transparenz des Systems. Sie erwarten die Möglichkeit, ihre Ergebnisse nachvollziehen zu können. Zu den Anwendungsfällen gehören die Anzeige der Teilnehmerliste sowie die Einsicht in die Rennergebnisse.

3.3. Systemarchitektur

Das Zeitnahmesystem umfasst mehrere Hauptkomponenten, wie in Abb. 1 dargestellt. Eine RFID-Antenne zur Erfassung von Transponderdaten, ein Python-Skript zur

Datenverarbeitung, eine MySQL-Datenbank zur Speicherung der erfassten Zeiten sowie eine Next.js-Webseite zur Anzeige der Ergebnisse und zur Verwaltung der Teilnehmer. Die modulare Architektur ermöglicht eine unabhängige Entwicklung jeder Komponente, die dennoch nahtlos in das Gesamtsystem integriert ist, um Skalierbarkeit und Wartbarkeit zu gewährleisten.

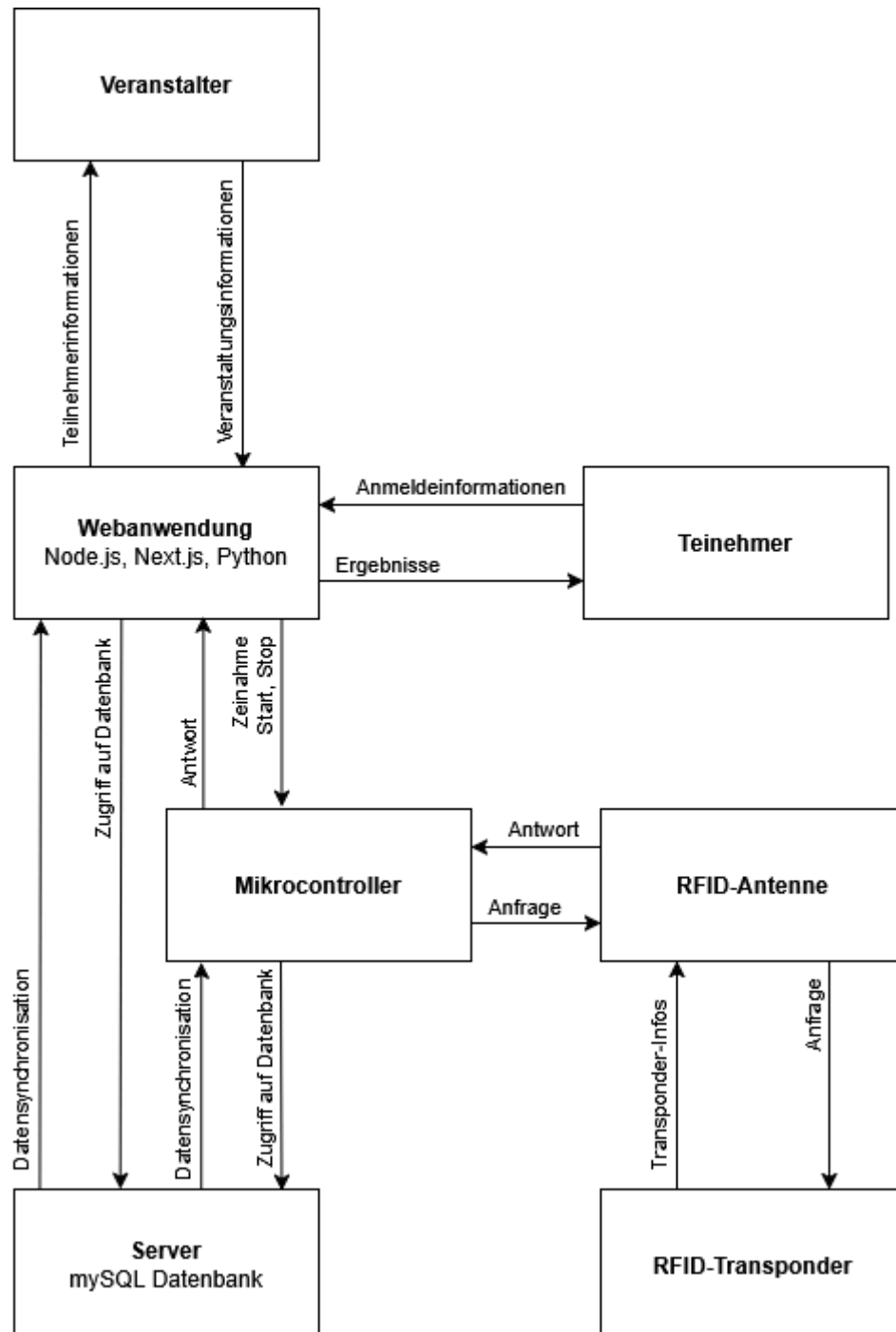


Abb. 1 Systemkontextdiagramm

3.4. Wireframes / Mockup

Das Ziel ist es, ein ansprechendes und benutzerfreundliches Design zu schaffen, das die Informationen auf eine klare und prägnante Weise präsentiert. Dabei müssen die Funktionalität, die Benutzerfreundlichkeit und die Skalierbarkeit berücksichtigt werden (s. [Ironhack_Barrierefreiheit]). Eine gute Datenvisualisierung zeichnet sich durch eine klare Struktur, eine einfache Sprache, eine klare visuelle Hierarchie und die Verwendung von relevanten Metriken aus. Es geht darum, das Layout, die Farben, die Schriftarten und die visuellen Elemente des Dashboards zu gestalten. Zur Planung der Benutzeroberfläche wurde eine Software zum Erstellen von sogenannten Wireframes verwendet. Wireframes sind digitale Skizzen, welche nur die rudimentärsten grafischen Anforderungen erfüllen. Das hat den Vorteil, dass schnell und einfach eine Struktur gefunden werden kann, ohne sich mit den besonderen grafischen Eigenheiten aufzuhalten. Das gesamte Zeitnahmesystem ist als Multiple-Document aufgebaut, die Seite "Veranstaltung bearbeiten" ist in Abb. 2 dargestellt. Ein weiteres Wireframe ist im Anhang A.2 dargestellt.

Info Anmeldung Teilnehmer Ergebnisse

Veranstaltung bearbeiten

Name

Veranstaltung 2

Webseite

https://www.veranstaltung2.de/

Beschreibung

bla

Altersklassen

Herren Elite Frauen Master 1 Master 2

Abb. 2 Wireframe Seite "Veranstaltung bearbeiten"

3.5. Datenhaltung

Im folgenden Entity-Relationship-Diagramm wird die Struktur der Datenbank und die Verknüpfungen zwischen den Tabellen visualisiert. Die Daten werden in mehreren Tabellen abgelegt: **events**, **groups**, **groupsmn**, **participants** und **tracking**. Die Verknüpfung zwischen den Tabellen erfolgt über eindeutige IDs, um eine präzise Zuordnung und einfache Abfrage der Daten zu ermöglichen. Die Tabelle **events** enthält die Veranstaltungen, die im System angelegt werden. Über die Tabelle **groupsmn** können den Veranstaltungen spezifische Altersklassen zugewiesen werden, die dann aus der Tabelle **groups** ausgewählt werden können. In der Tabelle **participants** werden die Daten der Teilnehmer gespeichert,

die sich für eine Veranstaltung anmelden. Die **tracking**-Tabelle erfasst die von der Zeitnahme ermittelten Daten und verknüpft diese über die Transponder-ID mit den Teilnehmern der jeweiligen Veranstaltung. Das ER-Diagramm verdeutlicht, wie die Daten zusammenhängen.

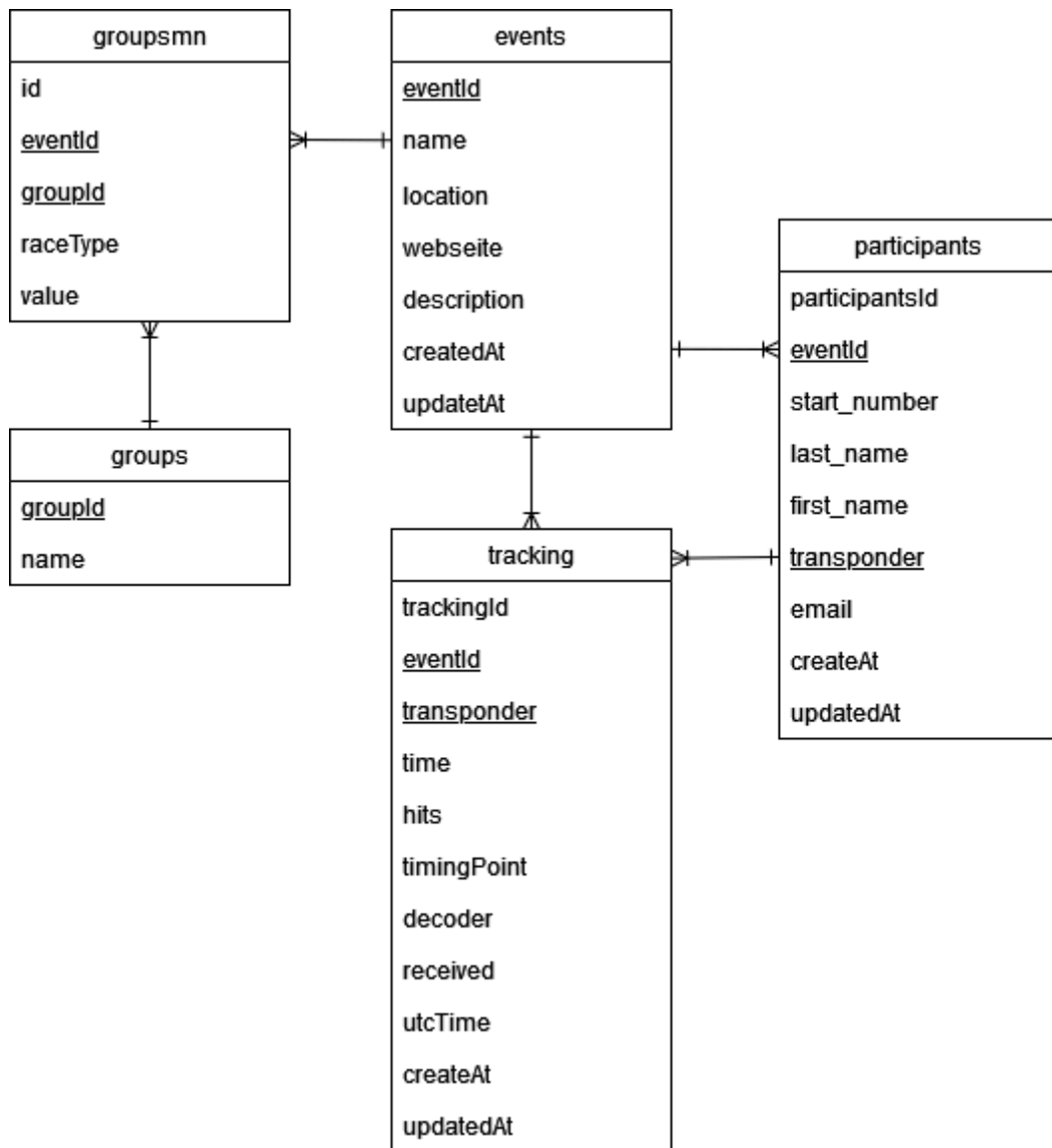


Abb. 3 ER-Diagramm

Im Beispiel sehen wir, dass der Veranstaltung 1 die Altersklassen „Männer Elite“ und „Frauen Elite“ zugewiesen wurden. Zwei Teilnehmer haben sich für diese Veranstaltung angemeldet und wurden über ihre Transponder während des Rennens erfasst. Die Daten der Zeitnahme aus der Tabelle **tracking** werden diesen Teilnehmern zugeordnet, wodurch eine genaue Zuordnung und Auswertung der Rennzeiten ermöglicht wird.

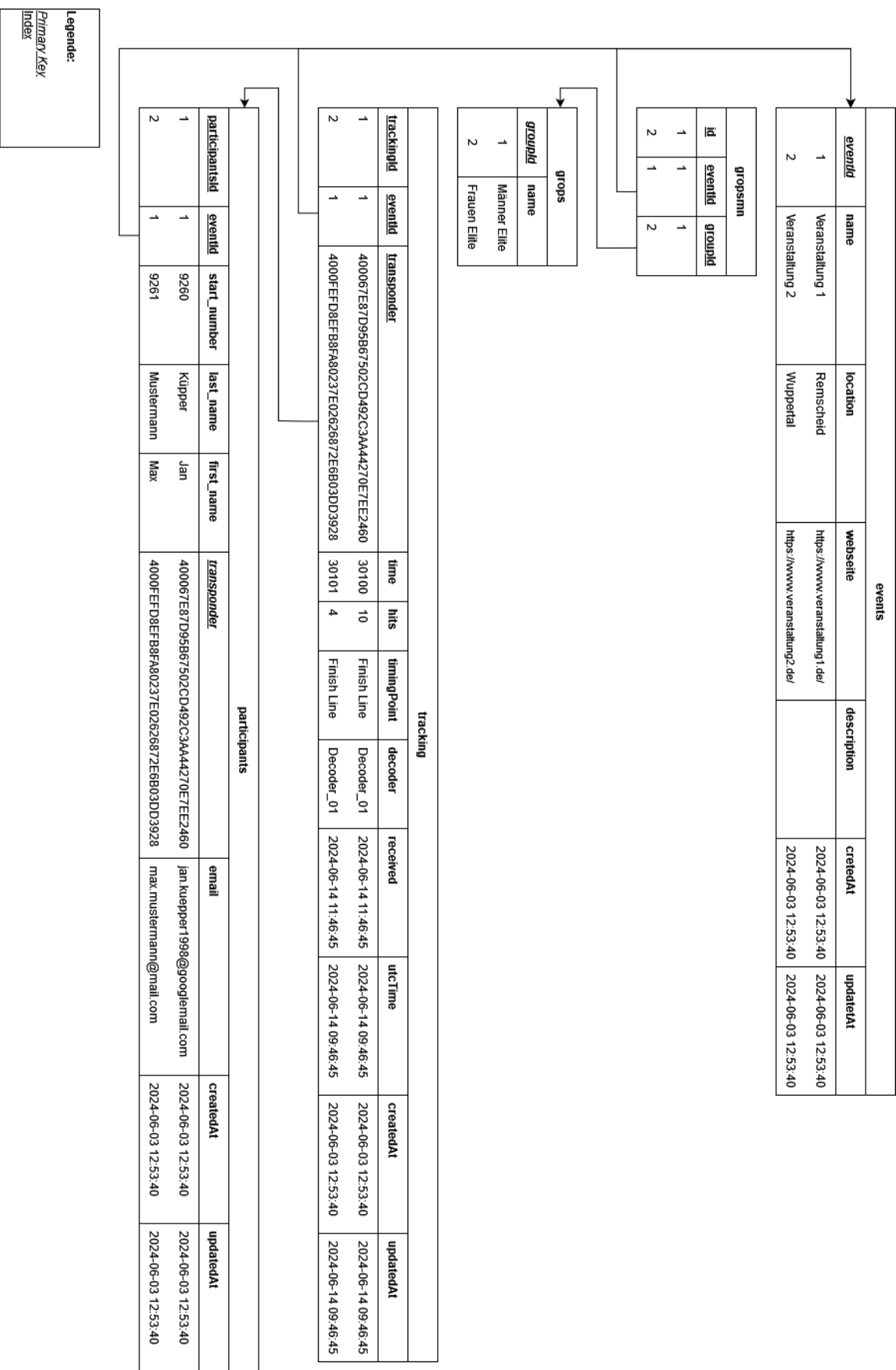


Abb. 4 Beispielhafte Datenhaltung

4. Entwicklung des Zeitnahme-System

Ein wesentlicher Bestandteil bei der Entwicklung des Zeitnahme-Systems ist die Wahl von Open-Source-Technologien, um die Flexibilität und Skalierbarkeit des Systems zu maximieren. Für die Entwicklung der Benutzeroberfläche wurde **React** in Verbindung mit dem **Next.js**-Framework eingesetzt und für das Design sowie die Benutzererfahrung kam **Material-UI** zum Einsatz. Diese Technologien bieten leistungsstarke Funktionalitäten und tragen gleichzeitig zur Minimierung externer Abhängigkeiten bei. Zur Verwaltung der Daten wird **MySQL** verwendet und für die Datenverarbeitung **Node.js** als auch **Python**.

4.1. Webapplikation

Die Webapplikation wurde mit **Node.js** und **Next.js** entwickelt, einem leistungsstarken React-Framework, das durch seine serverseitige Render-Funktionalität eine hohe Performance und schnelle Ladezeiten ermöglicht (s. [Vercel_Next.js]). Für die Gestaltung der Benutzeroberfläche wurde die **Material-UI-Bibliothek** (@mui/material) integriert, die eine Vielzahl an vorgefertigten Komponenten bereitstellt. Dies beschleunigt nicht nur die Entwicklung, sondern steigert auch die Wartbarkeit und sorgt für eine ansprechende und konsistente Benutzererfahrung (s. [Material_UI]).

Die **Next.js-Webseite** dient als Frontend des Zeitnahmesystems und übernimmt die Verwaltung der Teilnehmer sowie die Anzeige der Rennresultate. Bei der Implementierung wurde ein besonderer Fokus auf Benutzerfreundlichkeit und Funktionalität gelegt, um den Anwender bestmöglich zu unterstützen. Die Webseite bietet zwei Haupteingänge und ist so gestaltet, dass sie auf allen Endgeräten gut funktioniert sowie eine intuitive Navigation bietet. Zum einen gibt es den **Terminal-Zugang**, in dem Veranstaltungen angezeigt, Anmeldungen durchgeführt und Ergebnisse in Echtzeit abgerufen werden können. Zum anderen gibt es den **Admin-Zugang**. Dieser ermöglicht zusätzliche Funktionen wie das Erstellen und Bearbeiten von Veranstaltungen, das Verwalten von Teilnehmern, das Zuweisen von Startnummern und Transpondern sowie die Konfiguration der Zeitnahme. Außerdem gibt es ein Terminalfenster, in dem die Zeitnahme gestartet und Logdaten eingesehen werden können.

Für die Datenhaltung wurde **Sequelize**, ein leistungsstarkes ORM (Object-Relational Mapping) für Node.js, eingesetzt. **Sequelize** erleichtert die Interaktion mit der **MySQL**-Datenbank und ermöglicht effiziente und sichere Datenbankoperationen wie SELECT, INSERT und UPDATE. Das verbessert die Struktur des Codes und sorgt für eine hohe Wartbarkeit. Jede relevante Datenbanktabelle wurde mit einem entsprechenden Sequelize-Modell definiert, wobei eindeutige IDs verwendet werden, um klare Beziehungen zwischen den Tabellen herzustellen. Diese Struktur gewährleistet eine präzise Verwaltung der Rennzeiten und Teilnehmerdaten.

Alle genannten Technologien sind Open Source. **Next.js**, **Material-UI** und **Node.js** sind unter der MIT-Lizenz lizenziert verfügbar, was eine freie Nutzung, Anpassung und

Weiterverbreitung des Codes erlaubt. Diese Lizenz gewährleistet, dass Entwickler die Software weiter verbessern können, ohne auf rechtliche Einschränkungen zu stoßen.

4.2. Empfängereinheit

Die Empfängereinheit des Zeitnahme-Systems besteht aus einer RFID-Antenne mit einer eigenen Platine und einer seriellen USB-Schnittstelle als Empfänger. Diese Einheit arbeitet mit passiven Transpondern, die am Fahrrad der Teilnehmer befestigt werden. Da passive Transponder keine eigene Energiequelle besitzen, werden sie durch das elektromagnetische Feld der Antenne aktiviert.

RFID (Radio-Frequency Identification) ermöglicht die drahtlose Kommunikation zwischen dem RFID-Reader und den Transpondern mittels elektromagnetischer Wellen. In diesem System kommen passive **UHF-RFID-Transponder** (Ultra-High Frequency, 860-960 MHz) zum Einsatz, da sie sich besonders für hohe Geschwindigkeiten und große Erfassungsreichweiten eignen. Die Aktivierung des Transponders erfolgt, sobald die RFID-Antenne ein hochfrequentes elektromagnetisches Feld aussendet, das den Transponder mit Energie versorgt. Sobald sich ein Transponder in Reichweite der Antenne befindet, wird er induktiv aufgeladen und aktiviert (s. [XTremeGN_RFID]).

Nach der Aktivierung überträgt der Transponder seine eindeutige ID sowie weitere gespeicherte Daten über Backscatter-Modulation zurück an die Antenne. Dabei reflektiert der Transponder das ankommende Signal und moduliert es mit seinen eigenen Informationen. Die Antenne empfängt das rückgestrahlte Signal, verstärkt es und leitet es an den RFID-Reader weiter.

Sobald ein Teilnehmer die Ziellinie überquert, erfasst die Antenne den Transponder und überträgt die erfassten Daten über die serielle USB-Schnittstelle an ein Python-Skript. Das **Python**-Skript stellt das zentrale Steuerungselement des Zeitnahme-Systems dar. Es verarbeitet die empfangenen Daten und speichert diese anschließend in der **MySQL**-Datenbank. Eine besondere Herausforderung bei der Entwicklung war es, die Datenmengen effizient zu handhaben, da ein Transponder beim Überqueren der Ziellinie unter Umständen mehrfach erfasst werden kann. Das Skript wurde so optimiert, dass diese Mehrfacherfassungen erkannt und korrekt verarbeitet werden.

Das Python-Skript ist für den kontinuierlichen Betrieb und die effiziente Verarbeitung großer Datenmengen ausgelegt. Beim Start liest es die Konfigurationsdaten für das jeweilige Rennen ein und kommuniziert über die serielle Schnittstelle mit der RFID-Hardware. Die Konfigurationsdaten wie Event-ID, Timing-Point und Decoder-ID werden aus einer JSON-Datei geladen, um eine flexible Anpassung an verschiedene Rennformate zu ermöglichen. Die erfassten Zeitdaten werden strukturiert in den entsprechenden Tabellen der Datenbank abgelegt. Eindeutige IDs garantieren eine korrekte Verknüpfung zwischen den verschiedenen Tabellen, wie in Kapitel 3.5. beschrieben, was eine genaue und zuverlässige Zeitnahme ermöglicht, selbst bei großen Veranstaltungen mit vielen Teilnehmern.

Python wird unter der Python Software Foundation License vertrieben und **MySQL** ist unter der GNU General Public License (GPL) verfügbar. Beide Lizenzen erlauben die freie Nutzung und Weitergabe der Technologien. Wird die Software weiterentwickelt, müssen abgeleitete Werke ebenfalls unter der GPL-Lizenz veröffentlicht werden, um die Freiheit des Codes zu wahren und sicherzustellen, dass alle Verbesserungen unter den gleichen offenen Bedingungen verfügbar bleiben.

5. Nachhaltige Informationstechnologie

Bei der Entwicklung des System wurden verschiedene Prinzipien wie Konsistenz, Effizienz und Suffizienz berücksichtigt, um gleichzeitig die langfristige Funktionsfähigkeit und Skalierbarkeit des System zu gewährleisten. Darüber hinaus wird der Fokus auf die Wahl von Open-Source-Lizenzen, den Schutz personenbezogener Daten und die Sicherheit der Anwendung gelegt, um sowohl die rechtlichen als auch die ethischen Anforderungen der Nachhaltigkeit zu erfüllen.

5.1. Prinzipien der Nachhaltigkeit

Die Integration von Nachhaltigkeitsprinzipien in die Entwicklung des Zeitnahmesystems ist ein zentraler Aspekt des Projekts. Ziel ist es, Ressourcen effizient zu nutzen, die Lebensdauer der eingesetzten Technologien zu maximieren und die Umweltauswirkungen des Systems zu minimieren. Dabei stehen die Prinzipien Konsistenz, Effizienz und Suffizienz im Mittelpunkt.

Konsistenz bedeutet, Technologien und Materialien so auszuwählen, dass sie langfristig verfügbar und wartbar bleiben. Im Zeitnahmesystem wird dies durch die Nutzung von Open-Source-Technologien und eine modulare Architektur erreicht, die eine Wiederverwendbarkeit der Softwarekomponenten gewährleistet. Die eingesetzten RFID-Hardware-Komponenten sind langlebig und können über viele Rennveranstaltungen hinweg verwendet werden. Darüber hinaus erlaubt die modulare Systemarchitektur eine einfache Integration zukünftiger Upgrades, ohne bestehende Ressourcen zu verschwenden.

Effizienz zeigt sich im geringen Energieverbrauch und der optimierten Leistung des Systems. Die stromsparenden Transponder in Kombination mit Python für die Datenverarbeitung und MySQL für die Datenspeicherung ermöglichen eine hohe Leistung bei gleichzeitig niedrigem Energieeinsatz. Dies reduziert nicht nur die Betriebskosten, sondern auch die CO₂-Bilanz der eingesetzten Technologien.

Das Zeitnahmesystem umfasst ausschließlich die Funktionen, die für einen reibungslosen Ablauf von Radrennen unverzichtbar sind. Diese gezielte Fokussierung minimiert unnötigen System-Overhead, reduziert die Anforderungen an die Hardware und erhöht die Benutzerfreundlichkeit. Gleichzeitig werden die benötigten Ressourcen auf das Wesentliche beschränkt, um Verschwendung zu vermeiden.

Durch die konsequente Umsetzung dieser Prinzipien wird nicht nur die Umweltfreundlichkeit des Systems gefördert, sondern auch dessen wirtschaftliche und technische Nachhaltigkeit gestärkt. Das Ergebnis ist ein System, das den Anforderungen heutiger und zukünftiger Veranstalter gerecht wird, indem es Wiederverwendbarkeit, Energieeffizienz und eine schlanke Architektur in den Vordergrund stellt.

5.2. Lizenzen und Entscheidung für Open Source

Die Entscheidung, das Zeitnahmesystem unter Nutzung von Open-Source-Technologien zu entwickeln, ist gezielt darauf ausgerichtet, Nachhaltigkeit, Flexibilität und eine breite Zugänglichkeit zu fördern. Dabei wurde insbesondere die MIT-Lizenz gewählt, da sie nahezu uneingeschränkte Nutzungs- und Anpassungsmöglichkeiten bietet. Diese Lizenz erlaubt es Entwicklern, den Code frei zu modifizieren, weiterzugeben und in neuen Projekten zu verwenden, ohne dabei rechtliche Hürden überwinden zu müssen. Dies fördert die Weiterentwicklung durch eine breite Entwicklergemeinschaft und sorgt dafür, dass das System langfristig aktuell bleibt (s. [Härting_MIT]).

Für das Zeitnahmesystem bietet die Nutzung von Open-Source-Technologien wie Python, MySQL und Next.js erhebliche Vorteile. Sie ermöglichen nicht nur eine kostenfreie Nutzung, sondern gewährleisten auch eine hohe Flexibilität bei der Anpassung an spezifische Anforderungen von Veranstaltern. Beispielsweise kann die bestehende Architektur durch eigene Erweiterungen ergänzt werden, um zusätzliche Rennformate oder Funktionen zu unterstützen.

Ein weiterer Vorteil der Open-Source-Lösung ist die Transparenz. Der offene Quellcode ermöglicht eine gründliche Überprüfung, was Sicherheitsrisiken minimiert und die Zuverlässigkeit des Systems erhöht. Besonders im Hinblick auf Datenschutz und die Verarbeitung sensibler Teilnehmerdaten ist dies von großer Bedeutung. Veranstalter können sicherstellen, dass keine versteckten Sicherheitslücken oder unerwünschten Datenabflüsse existieren.

Darüber hinaus trägt die Wahl von Open Source zur Nachhaltigkeit des Projekts bei. Die Lebensdauer des Systems verlängert sich, da selbst nach Abschluss des Projekts das System von anderen Entwicklern weitergeführt und angepasst werden kann, was den langfristigen Nutzen maximiert.

Zusammenfassend ermöglicht die Entscheidung für Open Source eine kosteneffiziente, flexible und nachhaltige Entwicklung des Zeitnahmesystems, die sowohl den Veranstaltern als auch der Open-Source-Community zugutekommt. Die Lizenzwahl und die Nutzung etablierter Open-Source-Technologien stellen sicher, dass das System anpassungsfähig, sicher und zukunftsfähig bleibt.

5.3. Datenschutz und Sicherheit

Der Schutz der Daten von Teilnehmern und Veranstaltern sowie die Sicherheit der Anwendung spielen eine zentrale Rolle bei der Entwicklung des Zeitnahmesystems. Da das System persönliche Informationen wie Namen verarbeitet, sind umfassende Maßnahmen erforderlich, um den Datenschutz zu gewährleisten und den Missbrauch der Daten zu verhindern.

Bei der Entwicklung wurde großer Wert auf die Minimierung von potenziellen Angriffspunkten gelegt. Beispielsweise wird das System in einer isolierten Serverumgebung

betrieben, um externe Zugriffe zu kontrollieren. Die Daten werden ausschließlich vor Ort verarbeitet .

Durch diese Maßnahmen wird gewährleistet, dass das Zeitnahmesystem den Anforderungen an Datenschutz und Sicherheit entspricht. Veranstalter können darauf vertrauen, dass die Daten ihrer Teilnehmer geschützt sind, während gleichzeitig eine zuverlässige und transparente Zeitmessung ermöglicht wird. Diese Kombination aus Datensicherheit und Nutzerfreundlichkeit trägt entscheidend dazu bei, das Vertrauen in das System zu stärken.

5.4 Qualitätssicherung

Die Qualitätssicherung und die Tests des Zeitnahmesystems spielen eine zentrale Rolle, um die Funktionalität, Zuverlässigkeit und Benutzerfreundlichkeit des Systems sicherzustellen. Bereits während der Entwicklung wurden Maßnahmen ergriffen, um mögliche Schwachstellen frühzeitig zu identifizieren und zu beheben.

Ein wesentlicher Bestandteil der Qualitätssicherung war die Versionierung des Codes. Neue Funktionen und Anpassungen wurden in separaten Branches entwickelt und gründlich getestet, bevor sie in den Hauptentwicklungszweig integriert wurden. Diese Vorgehensweise gewährleistete, dass nur stabiler und getesteter Code in das Gesamtsystem aufgenommen wird. Zusätzlich wird auf eine klare Strukturierung und umfangreiche Dokumentation des Codes geachtet, um die Wartbarkeit und zukünftige Weiterentwicklung zu erleichtern.

Ein zentraler Schwerpunkt der Tests lag auf der Überprüfung des Zusammenspiels der verschiedenen Systemkomponenten, der RFID-Hardware, der Python-Datenverarbeitung und der MySQL-Datenbank. Ziel war es, sicherzustellen, dass alle Teile nahtlos miteinander kommunizieren und die erfassten Zeiten präzise verarbeitet und zuverlässig gespeichert werden.

Die RFID-Hardware wurde unter realistischen Bedingungen getestet, um ihre Lesegenauigkeit, Reichweite und Zuverlässigkeit zu bewerten. Dabei wurden verschiedene Szenarien simuliert, darunter hohe Geschwindigkeiten bei der Überquerung der Ziellinie mit bis zu 40 km/h, um zu prüfen, ob die Transponder auch bei kurzen Signalübertragungszeiten korrekt erfasst werden. In 90% der Fälle erfolgte die Registrierung fehlerfrei, während kleinere Abweichungen hauptsächlich durch extrem kurze Kontaktzeiten zwischen Transponder und Antenne entstanden. Ein weiterer Test untersuchte, wie zuverlässig mehrere Teilnehmer erfasst werden, wenn sie gleichzeitig die Ziellinie überqueren. Hier konnte das System in rund 80 % der Fälle alle Transponder korrekt registrieren, wobei Mehrfacherfassungen durch das nachgelagerte Python-Skript herausgefiltert wurden.

Die Python-Datenverarbeitung wurde insbesondere auf ihre Effizienz bei der Echtzeitverarbeitung großer Datenmengen getestet. In simulierten Rennen konnte das Skript alle Daten ohne Verzögerung verarbeiten, während doppelte Erfassungen innerhalb von kürzester Zeit zuverlässig erkannt und bereinigt wurden.

Auch die MySQL-Datenbank wurde intensiven Belastungstests unterzogen, um ihre Leistungsfähigkeit und Datenkonsistenz zu gewährleisten. Dabei wurden Einträge in kurzen Zeitabständen hinzugefügt, um die Performance zu bewerten. Selbst unter hoher Last blieben die Abfragezeiten stabil. Zusätzlich wurden die gespeicherten Transponder-IDs mit den tatsächlich erfassten Rennzeiten abgeglichen, um sicherzustellen, dass keine fehlerhaften oder doppelten Einträge vorhanden waren. Diese Tests bestätigten die hohe Stabilität und Genauigkeit der Datenverarbeitung.

Die Webanwendung, die auf Next.js basiert, wurde hinsichtlich ihrer Benutzerfreundlichkeit und Funktionalität durch unterschiedliche Benutzer erprobt. Die wichtigsten Funktionen, darunter die Verwaltung von Veranstaltungen, die Registrierung von Teilnehmern und die Anzeige der Rennergebnisse, wurden gezielt getestet. Das Feedback zeigte, dass die meisten Funktionen intuitiv nutzbar waren, wobei kleinere Anpassungen an der Navigation vorgenommen wurden, um die Nutzerfreundlichkeit weiter zu verbessern. Um die Belastbarkeit der Webanwendung zu prüfen, wurden zudem Stresstests mit mehreren gleichzeitigen Zugriffen durchgeführt. Dabei blieb das System stabil.

Zusammenfassend wurden durch die umfassenden Maßnahmen zur Qualitätssicherung und die gezielten Tests die Funktionalität und Zuverlässigkeit des Zeitnahmesystems sichergestellt. Veranstalter können darauf vertrauen, dass das System unter realen Bedingungen einwandfrei arbeitet, während die Teilnehmer von einer präzisen und transparenten Zeitmessung profitieren. Diese Qualitätssicherungsstrategie bildet die Grundlage für ein robustes, skalierbares und zukunftsfähiges Zeitnahmesystem.

6. Fazit und Ausblick

Zusammenfassend wurde ein Zeitnahme-System entwickelt, das den Anforderungen sowohl der Veranstalter als auch der Teilnehmer gerecht wird. Die Anwendung ermöglicht eine präzise, zuverlässige und benutzerfreundliche Zeitmessung und bildet eine solide Grundlage für zukünftige Erweiterungen.

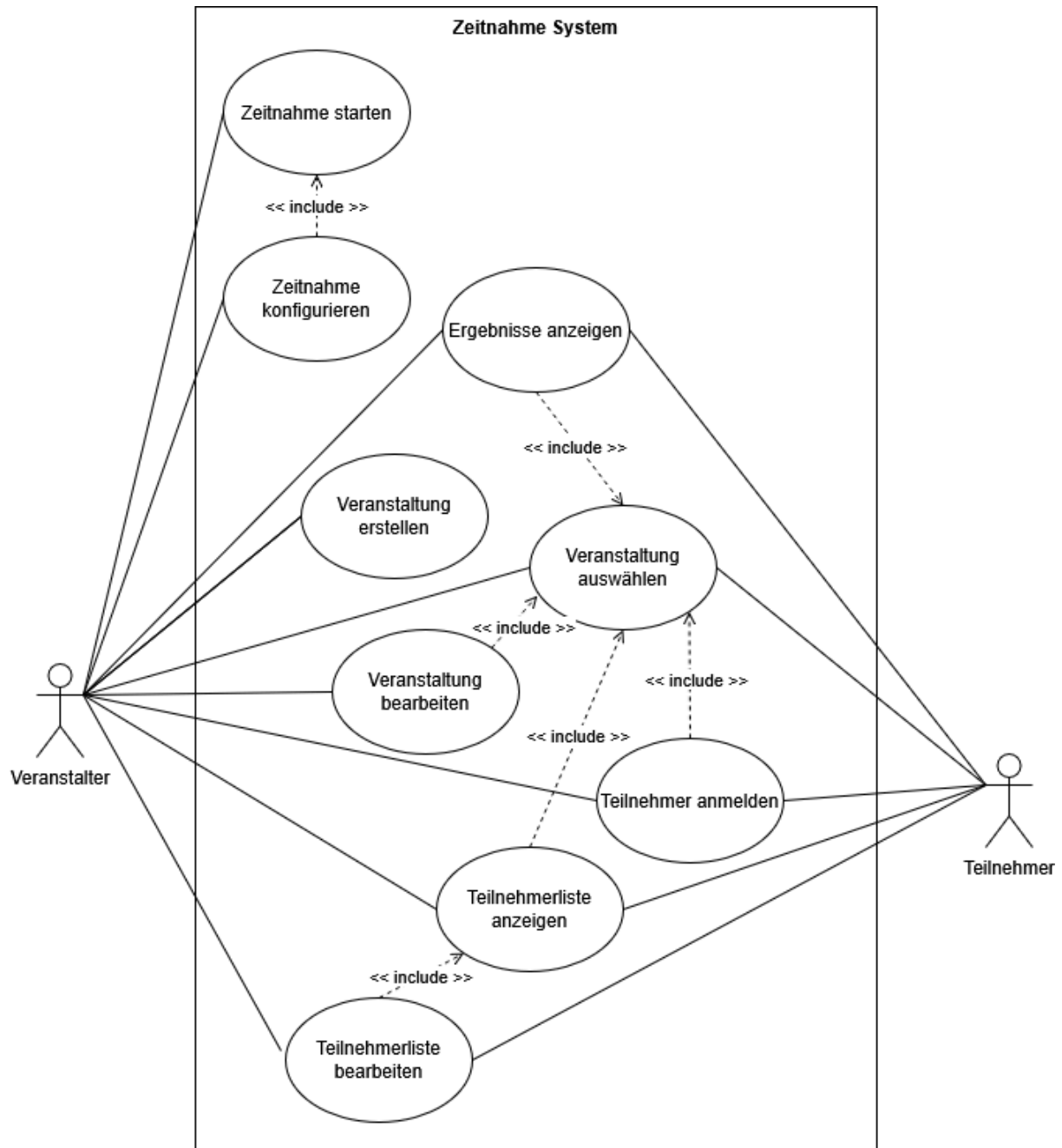
Für die Weiterentwicklung des Systems bieten sich vielfältige Möglichkeiten. Das Hauptziel für die Zukunft liegt in der weiteren Optimierung der Benutzerfreundlichkeit sowie der Steigerung der Leistungsfähigkeit des Systems. Besonders wertvolles Feedback von Anwendern wird hierbei genutzt, um konkrete Verbesserungen vorzunehmen.

Ein wichtiger Schwerpunkt wird auf der Optimierung für Rennen liegen, bei denen Start und Ziel an unterschiedlichen Orten liegen. Diese Anforderung ist besonders relevant für Zeitfahren mit Einzelstarts. Geplant ist, mehrere Zeitnahme-Punkte zu implementieren, sowohl am Start als auch im Ziel, um eine präzise und zuverlässige Zeitmessung über die gesamte Strecke hinweg sicherzustellen.

Durch diese Erweiterungen soll nicht nur die Leistungsfähigkeit des Systems weiter gesteigert werden, sondern auch die Unterstützung verschiedener Rennformate und Veranstaltungen verbessert werden. Die langfristige Vision ist es, ein umfassendes, flexibles und benutzerfreundliches Zeitnahme-System zu entwickeln, das sich an die unterschiedlichen Bedürfnisse und Herausforderungen moderner Radsportveranstaltungen anpasst.

7. Anhang

A.1 Anwendungsfalldiagramm



A.1 Wireframe Seite “Veranstaltungsübersicht”



Literatur- und Quellenverzeichnis

[Ironhack_Barrierefreiheit] Die Wichtigkeit der Barrierefreiheit im Web Design, Ironhack, URL: <https://www.ironhack.com/de/blog/die-wichtigkeit-der-barrierefreiheit-im-web-design> (Abgerufen: 03. Januar 2025, 10:00 UTC)

[Vercel_Next.js] What is Next.js, Vercel, URL: <https://nextjs.org/docs> (Abgerufen: 10. Januar 2025, 09:00 UTC)

[Material_UI] Material UI - Overview, Material UI SAS, URL: <https://mui.com/material-ui/getting-started/> (Abgerufen: 10. Januar 2025, 10:00 UTC)

[XTremeGN_RFID] Was ist RFID?, XeTremGN, URL: <https://www.rfid-basis.de/rfid-technik.html> (Abgerufen: 17. Januar 2025, 10:30 UTC)

[Härting_MIT] Software unter MIT Lizenz rechtssicher verwenden, Härting, URL: <https://haerting.de/wissen/software-unter-mit-lizenz-rechtssicher-verwenden/> (Abgerufen: 20. Januar 2025, 17:00 UTC)

Abkürzungsverzeichnis

JSON	“Java Script Object Notation”
MySQL	“My-Structured Query Language”
RFID	“Radio Frequency Identification”
UHF	“Ultra-High Frequency”

Abbildungsverzeichnis

Abb. 1 Systemkontextdiagramm	6
Abb. 2 Wireframe Seite "Veranstaltung bearbeiten"	7
Abb. 3 ER-Diagramm	8

Eidesstattliche Erklärung

Ich versichere, dass ich die Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Die Regelung der geltenden Prüfungsordnung zu Versäumnis, Rücktritt, Täuschung und Ordnungsverstoß habe ich zur Kenntnis genommen.

Diese Arbeit hat in gleicher oder ähnlicher Form keiner Prüfungsbehörde vorgelegen.

Remscheid, den 21.03.2025



Jan Küpper